



アーキテクチャ図

AWS でのウェブアプリケーションアーキテクチャ



AWS でのウェブアプリケーションアーキテクチャ: アーキテクチャ図

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

ウェブアプリケーションアーキテクチャ i

AWS でのウェブアプリケーションアーキテクチャ 1

詳細情報 2

図の履歴 2

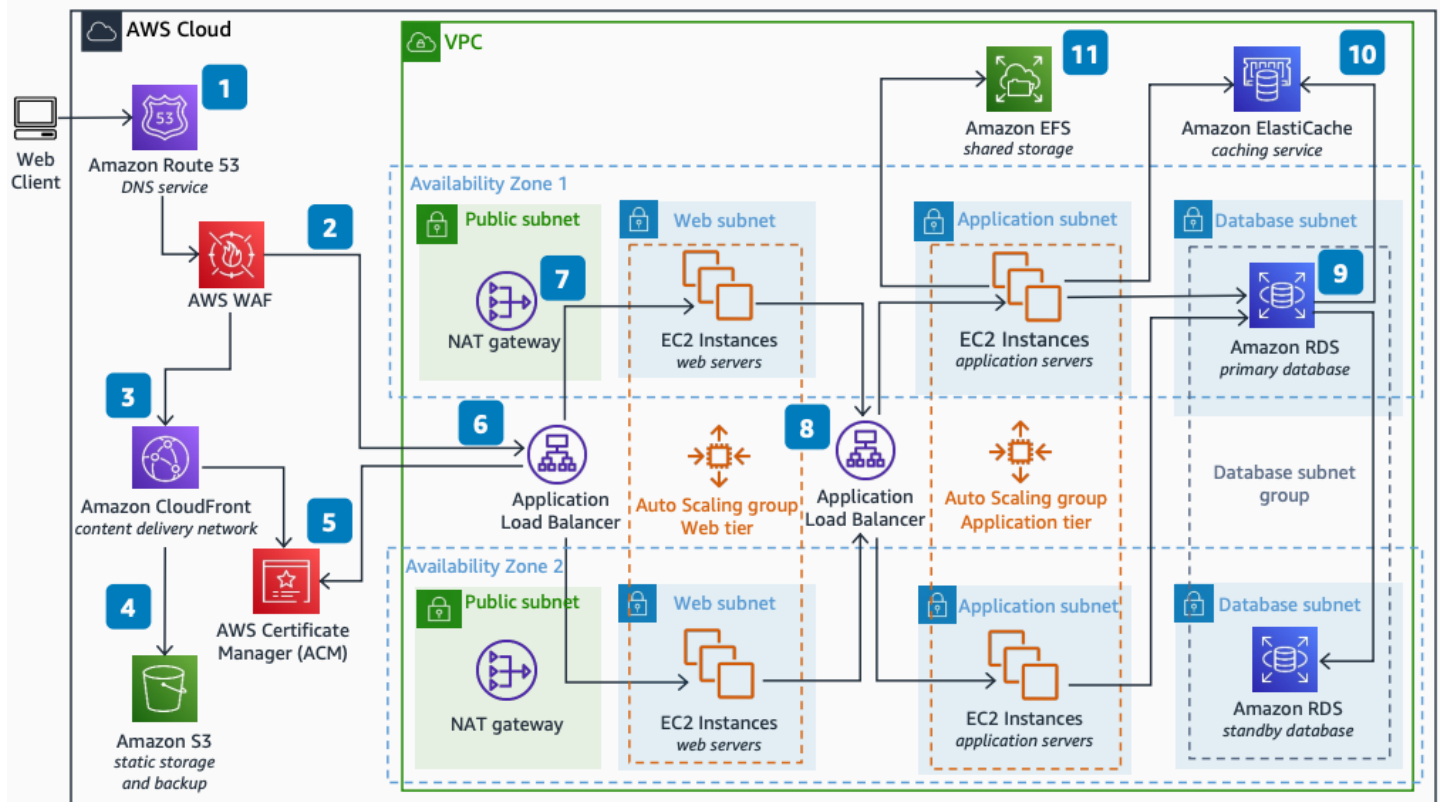
..... iv

AWS でのウェブアプリケーションアーキテクチャ

公開日: 2021 年 11 月 19 日 ([図の履歴](#))

このアーキテクチャは、でクラシックウェブアプリケーションをホストする方法を示しています AWS。高可用性には複数のアベイラビリティゾーン、トラフィック分散には Application Load Balancer、データベース、キャッシュ、共有ストレージにはマネージドサービスを使用します。

AWS でのウェブアプリケーションアーキテクチャ



次の手順では、アーキテクチャについて説明します。

1. Route [Route 53](#) します。
2. [Amazon CloudFront](#) などのコンテンツ配信ネットワーク (CDN) を使用して、静的コンテンツを配信する際のレイテンシーを短縮します。
3. [Amazon Simple Storage Service](#) を使用して、静的コンテンツとバックアップを保存します。
4. AWS WAF などのウェブアプリケーションファイアウォールを使用して、一般的なウェブエクспロイトからウェブアプリケーションを保護します。

5. AWS Certificate Manager (ACM) を使用して SSL 証明書の管理を簡素化します。
6. インターネット向け Application Load Balancer を使用して、複数のアベイラビリティーゾーンにまたがるウェブサーバーにウェブトラフィックを分散します。
7. 各パブリックサブネットに NAT ゲートウェイを使用して、プライベートサブネット内の [Amazon Elastic Compute Cloud](#) インスタンスがインターネットにアクセスできるようにします。
8. 内部 Application Load Balancer を使用して、複数のアベイラビリティーゾーンにまたがるアプリケーションサーバーにトラフィックを分散します。
9. [Amazon RDS](#) でデータベースレイヤーを実行して、データベース管理を簡素化します。
10. データベースアクセスパターンが読み取りが多い場合は、[Amazon ElastiCache](#) などのキャッシュレイヤーを活用します。
11. サーバーが共有ファイルにアクセスする場合は、Amazon Elastic File System (Amazon EFS) などの共有ストレージサービスの使用を検討してください。

詳細情報

詳細については、次のリソースを参照してください。

- [AWS アーキテクチャアイコン](#)
- [AWS アーキテクチャセンター](#)
- [AWS Well-Architected](#)

図の履歴

このリファレンスアーキテクチャ図の更新について通知を受け取るには、RSS フィードをサブスクライブします。

変更	説明	日付
初版発行	リファレンスアーキテクチャ図が最初に公開されました。	2021 年 11 月 19 日

 Note

RSS 更新をサブスクライブするには、使用しているブラウザで RSS プラグインが有効になっている必要があります。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。